

WCSCC Computational Systems Biology for Complex Human Disease

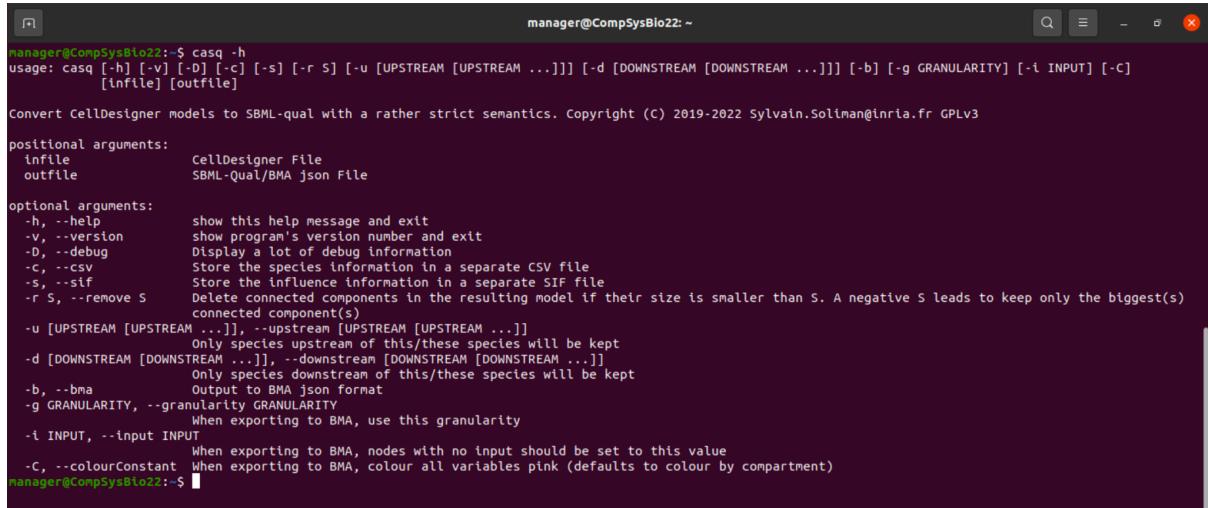
4-9 December 2022

Hands-on session with CaSQ

Instructors: Anna Niarakis & Sylvain Soliman

In this session, we are going to create an executable, SBML qual file from a CellDesigner diagram and use it to perform *in silico* simulations.

To do so, first, we will open the shell and type in the terminal the command: casq -h



A screenshot of a terminal window titled "manager@CompSysBio22:~". The window displays the usage information for the "casq" command. The text is as follows:

```
manager@CompSysBio22:~$ casq -h
usage: casq [-h] [-v] [-D] [-c] [-s] [-r S] [-u [UPSTREAM [UPSTREAM ...]]] [-d [DOWNSTREAM [DOWNSTREAM ...]]] [-b] [-g GRANULARITY] [-i INPUT] [-c]
            [infile] [outfile]

Convert CellDesigner models to SBML-qual with a rather strict semantics. Copyright (C) 2019-2022 Sylvain.Soliman@inria.fr GPLv3

positional arguments:
  infile           CellDesigner File
  outfile          SBML-Qual/BMA json File

optional arguments:
  -h, --help        show this help message and exit
  -v, --version     show program's version number and exit
  -D, --debug       Display a lot of debug information
  -c, --csv         Store the species information in a separate CSV file
  -s, --sif         Store the influence information in a separate SIF file
  -r S, --remove S Delete connected components in the resulting model if their size is smaller than S. A negative S leads to keep only the biggest(s)
                     connected component(s)
  -u [UPSTREAM [UPSTREAM ...]], --upstream [UPSTREAM [UPSTREAM ...]]
                     Only species upstream of this/these species will be kept
  -d [DOWNSTREAM [DOWNSTREAM ...]], --downstream [DOWNSTREAM [DOWNSTREAM ...]]
                     Only species downstream of this/these species will be kept
  -b, --bma         Output to BMA json format
  -g GRANULARITY, --granularity GRANULARITY
                     When exporting to BMA, use this granularity
  -i INPUT, --input INPUT
                     When exporting to BMA, nodes with no input should be set to this value
  -c, --colourConstant When exporting to BMA, colour all variables pink (defaults to colour by compartment)

manager@CompSysBio22:~$
```

Figure 1. Screenshot of a terminal and the options of the tool

As seen in Figure 1, CaSQ can produce a csv file that contains information about the species of the CellDesigner diagram and also the logical formulae of the regulations.

It can also produce SIF files (Simple Interaction Files) of two types that can be used in different tools. One apparent tool would be Cytoscape.

SIF files do not contain annotations, formulae or layout; they only have source targets, type of regulation and the target node.

Lastly, the major output file of CaSQ is an SBML qual file, that contains all information needed to perform dynamical modelling analyses and simulations.

Exercise 1: Create the files

Use as input file the xml file of the Apoptosis diagram always present in the repository and type the following code (adapted to your path settings):

```

manager@CompSysBio22:~$ casq -s course_data/Curation_Apoptosis_stable.xml course_data/Curation_Apoptosis_stable.sbml
manager@CompSysBio22:~$ ls -la course_data/
total 240
drwxrwxr-x 2 manager manager 4096 Nov 22 13:48 .
drwxr-xr-x 28 manager manager 4096 Nov 22 13:48 ..
-rw-rw-r-- 1 manager manager 901 Nov 22 13:48 Curation_Apoptosis_Apoptosis_stable_raw.sif
-rw-rw-r-- 1 manager manager 44106 Nov 22 13:48 Curation_Apoptosis_Apoptosis_stable.sbml
-rw-rw-r-- 1 manager manager 1329 Nov 22 13:48 Curation_Apoptosis_Apoptosis_stable.sif
-rw-rwx--- 1 manager manager 169511 Nov 3 12:11 Curation_Apoptosis_Apoptosis_stable.xml
-rw-rwx--- 1 manager manager 8284 Nov 3 12:11 xgml_to_gml.R
manager@CompSysBio22:~$ █

```

Figure 2. Example of the command that is needed to create the files

When you have successfully executed the command then in the corresponding folder (where your input file is located) you will see three different files:

📄 Curation_Apoptosis	16/11/2020 15:21	Fichier SBML	45 Ko
📄 Curation_Apoptosis	16/11/2020 15:21	Fichier SIF	2 Ko
📄 Curation_Apoptosis_raw	16/11/2020 15:21	Fichier SIF	1 Ko

Figure 3. Screenshot of the three different files that are created in the directory after the command execution.

Exercise 2. Open the files in Cytoscape

Import the obtained sif files and the sbml file in Cytoscape and discuss differences.

File- Import Network from file

Search and choose the SIF file

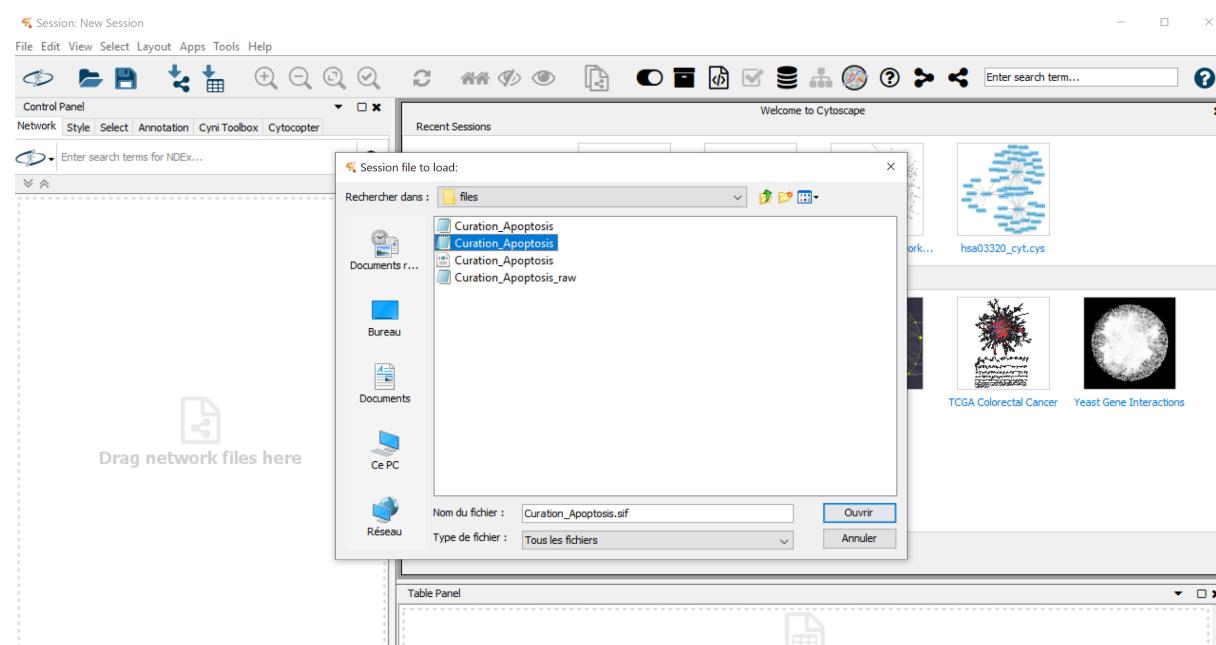


Figure 4. Import the CaSQ produced sif file to Cytoscape

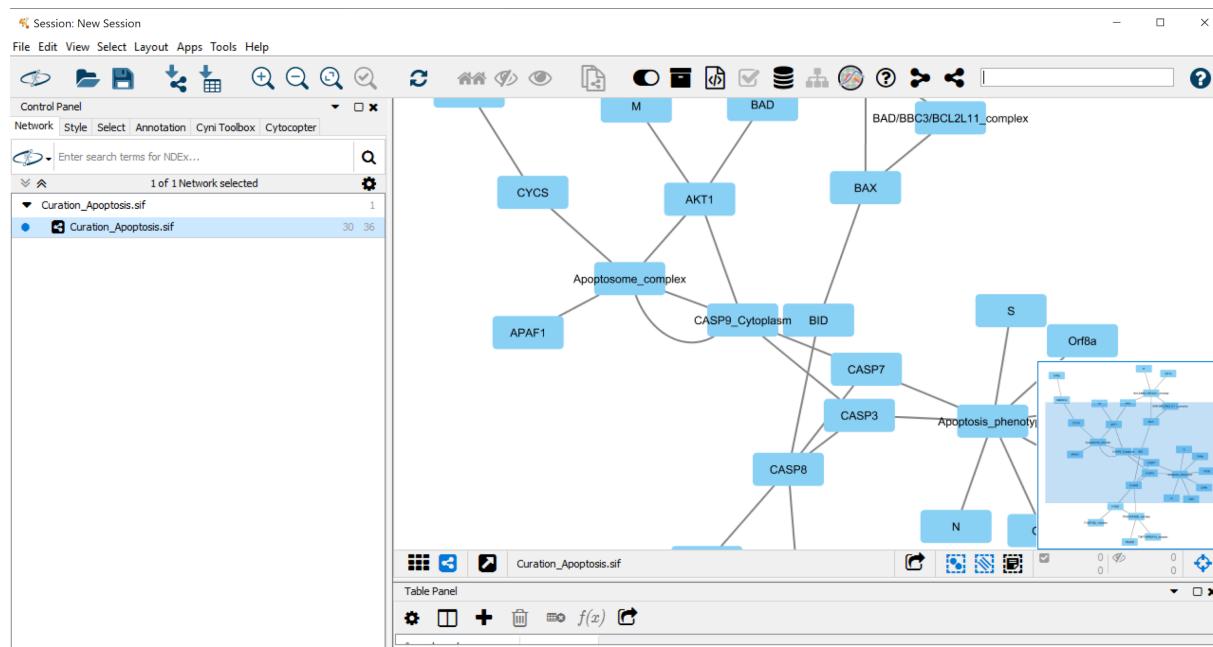


Figure 5. The obtained network in Cytoscape.

Then redo the same in the same session and open the raw SIF file:

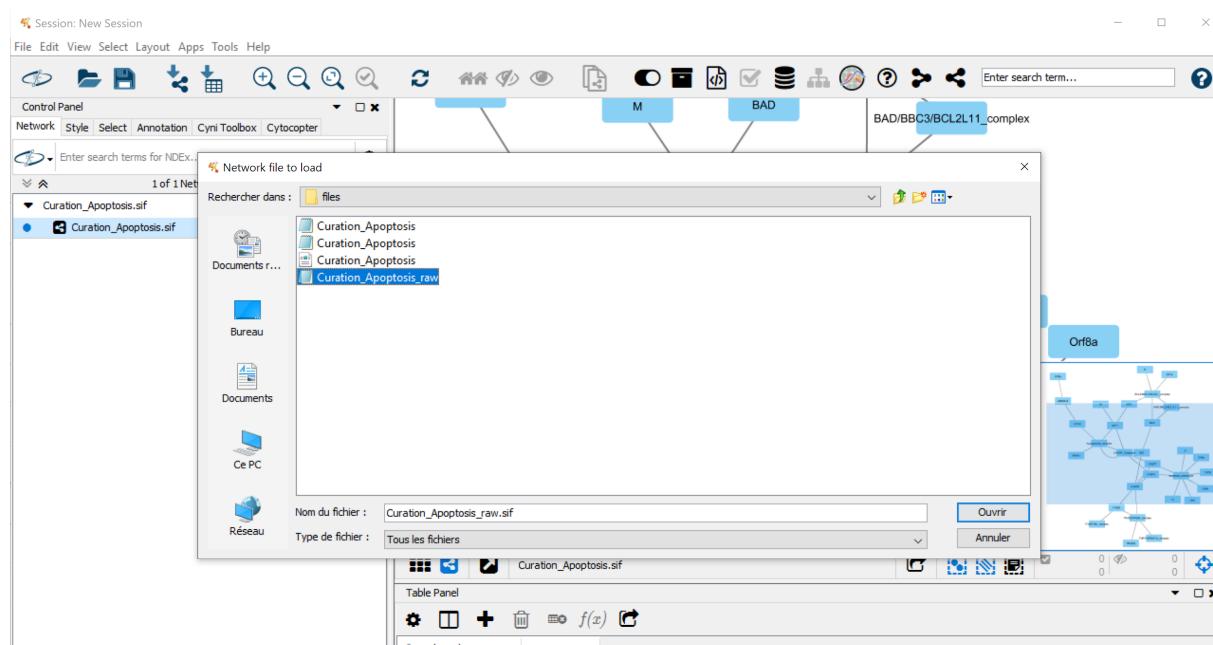


Figure 6. Import the raw SIF file in Cytoscape.

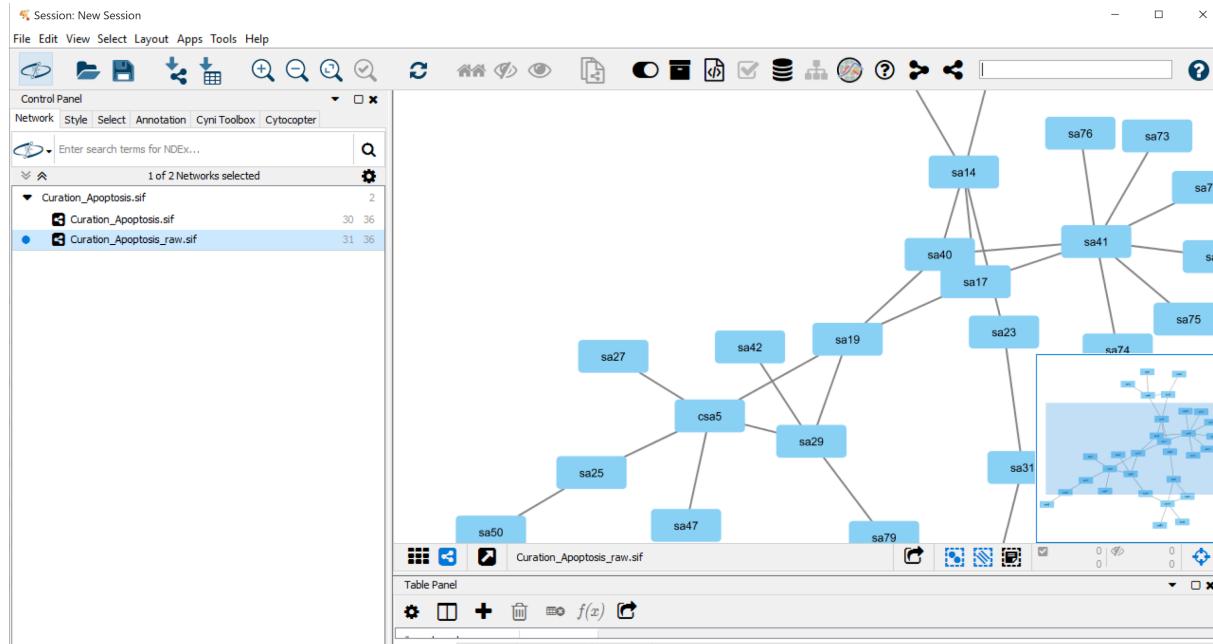


Figure 7. The obtained network in Cytoscape.

What do you observe regarding the names and also the layout, the size (number of interactions and number of nodes) regarding the networks of the two files?

Now open the sbml file in Cytoscape following the same mode
 File-Import network from file
 Choose the sbml file

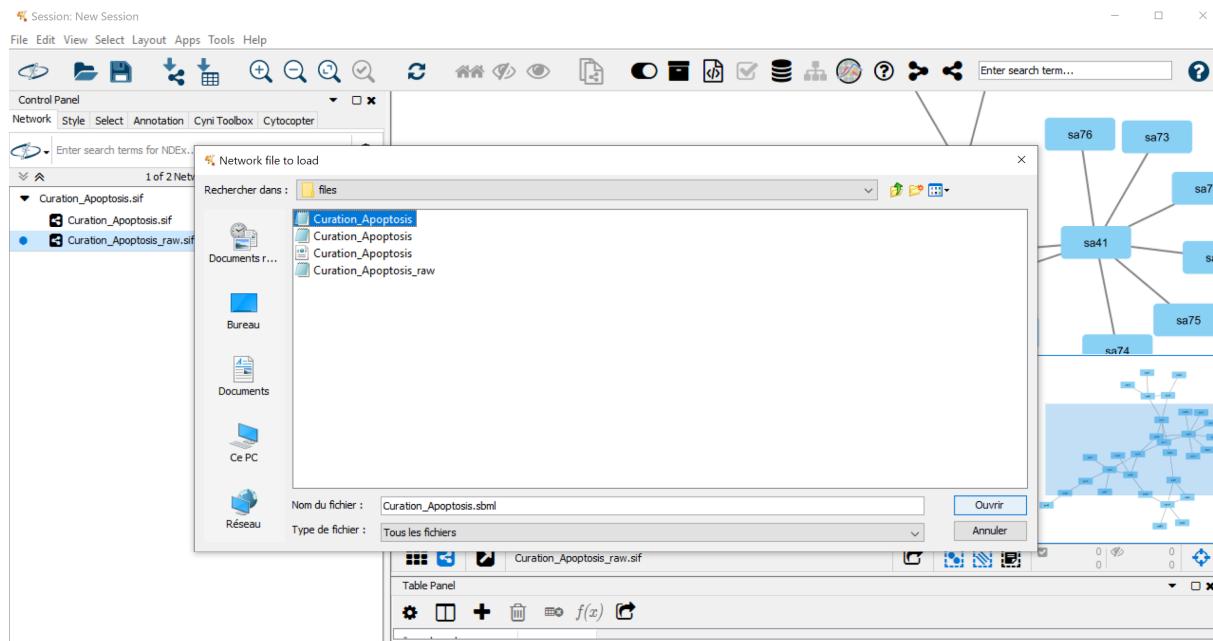


Figure 8. Import the sbml file as previously.

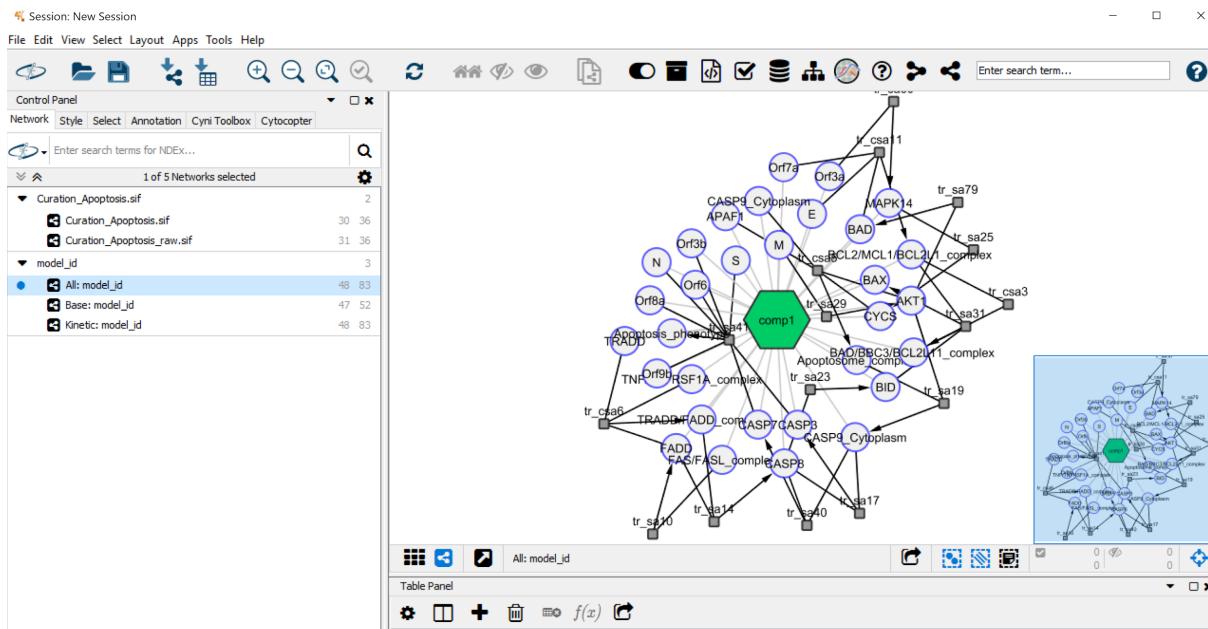


Figure 9. Imported network from the sbml file. Observe the three different networks in the left panel.

Now from the panel on the left choose the Base

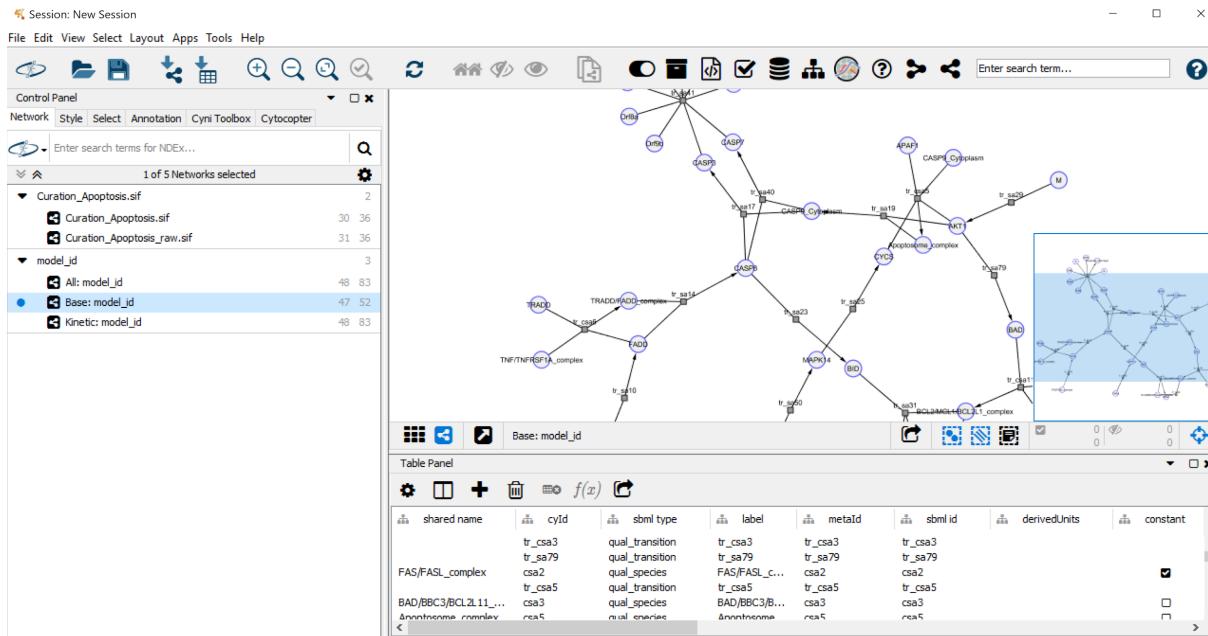


Figure 10. The Base model as seen in Cytoscape.

What do you observe? Why do you have a difference between the base and the all (kinetic) model?

Give your explanations.

Exercise 3. Import the SBML-qual file in Cell Collective

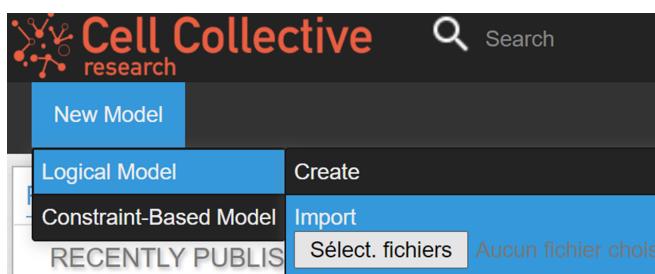


Figure 11. Click on New model, select Logical model, then Import then select from files.

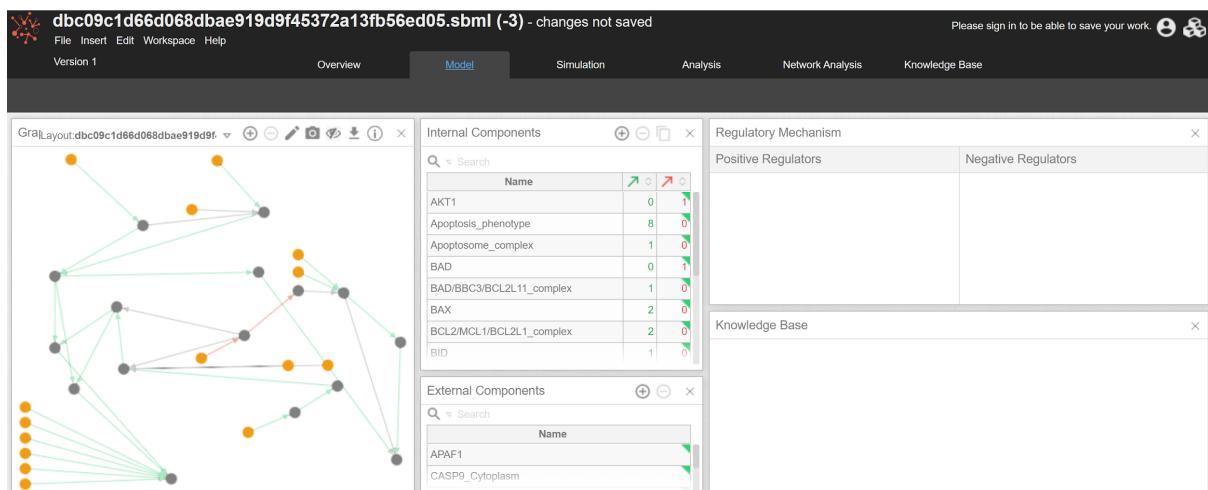


Figure 11. The model as seen in Cell Collective. The original layout of the diagram is retained.

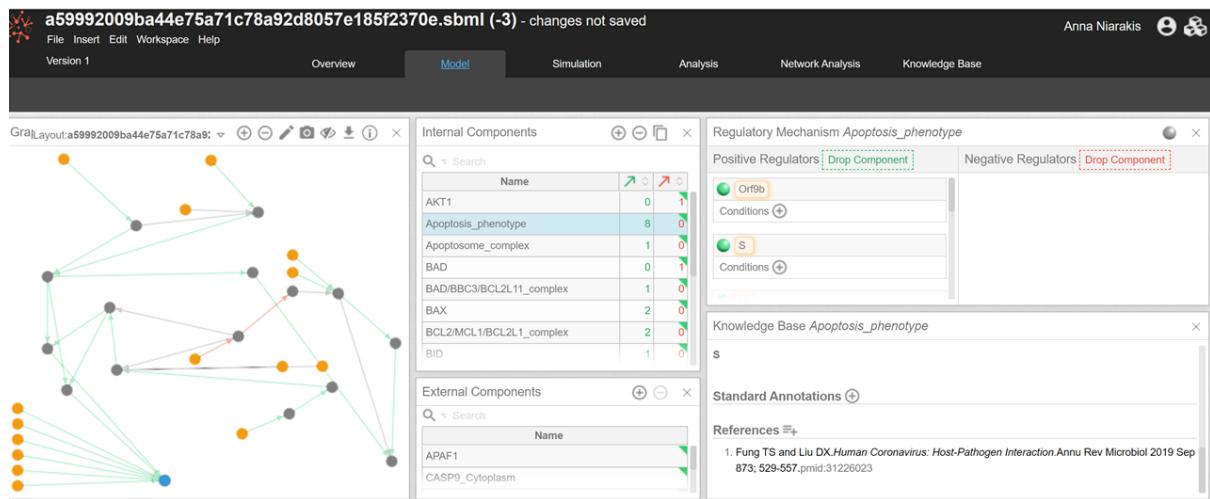


Figure 12. Click on the Apoptosis node. The references in the MIRIAM section of the original xml file are retained in the sbml qual file and can be retrieved.

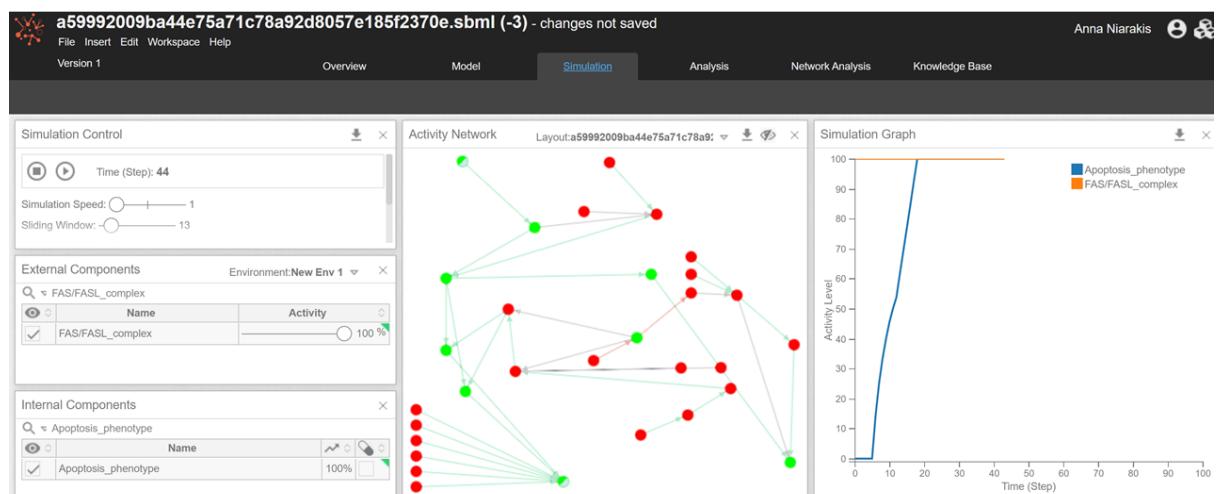
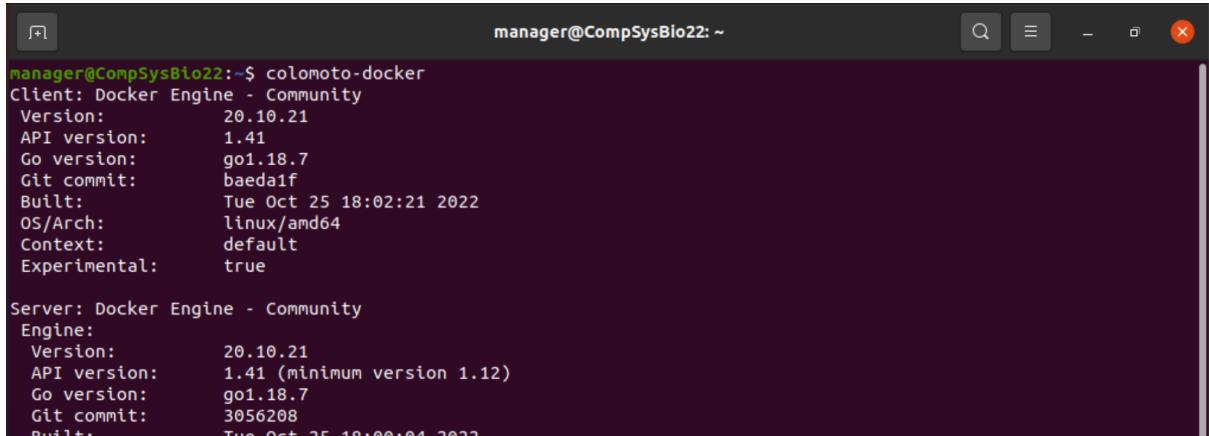


Figure 13. The model is fully executable and you can use it to perform simulations (more details on the Cell Collective tutorial that will follow).

Exercise 4. The CoLoMoTo notebook

Tape in your terminal the following command: colomoto-docker

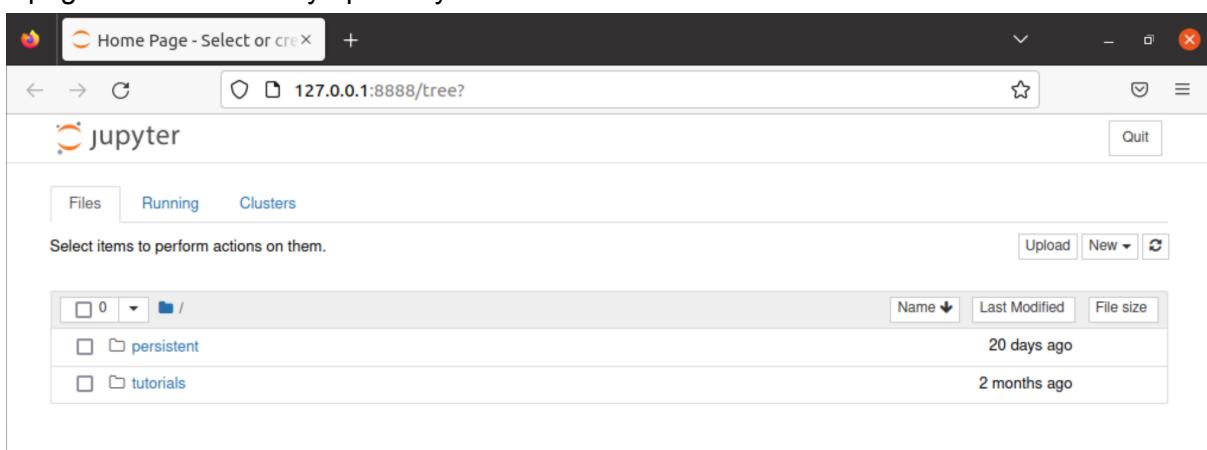


```
manager@CompSysBio22:~$ colomoto-docker
Client: Docker Engine - Community
 Version:          20.10.21
 API version:     1.41
 Go version:      go1.18.7
 Git commit:       baedaf
 Built:           Tue Oct 25 18:02:21 2022
 OS/Arch:         linux/amd64
 Context:          default
 Experimental:    true

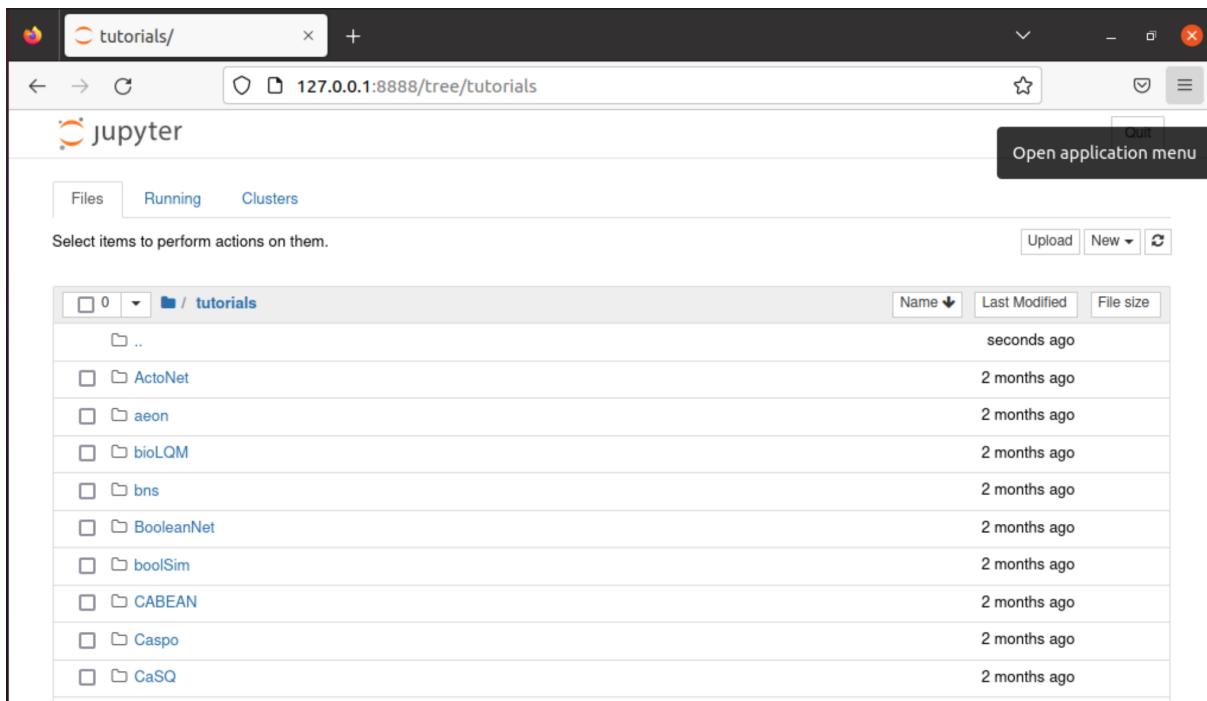
Server: Docker Engine - Community
 Engine:
  Version:          20.10.21
  API version:     1.41 (minimum version 1.12)
  Go version:      go1.18.7
  Git commit:       3056208
  Built:           Tue Oct 25 18:00:04 2022
```

Figure 14. Screenshot of the terminal after you tap the command and call the colomoto-docker image.

A page will automatically open in your browser. Click on the folder tutorials



Then select the folder CaSQ



Then select the jupyter notebook



Figure 15. Screenshots showing the steps to find the casq tutorial.

You will now perform your analysis in a reproducible and automated way: To execute the commands you need to click on the black arrowheads as shown in Figure 16:

```

Create an executable model with CaSQ

Entrée [1]: M import casq.celldesigner2qual as casq
from colomoto_jupyter import tabulate

# debug messages are enabled by default
casq.logger.disable("casq")

Entrée [2]: M # Load and simplify a cell designer map
info, width, height = casq.read_celldesigner("Apoptosis_VS_SSA_AN.xml")
casq.simplify_model(info)

# Write the SBML file
casq.write_qual("Apoptosis_VS_SSA_AN.sbml", info, width, height)

```

Figure 16. Execute the commands to obtain an sbml file

In the next two steps, you are going to import the sbml qual file to GINsim, another software for logical modelling and simulations and you will visualize your model

Load and view the model in GINsim

```
Entrée [3]: M import biolqm
import ginsim

Entrée [4]: M m = biolqm.load("Apoptosis_VS_SSA_AN.sbml")
m = biolqm.sanitize(m)

lrg = biolqm.to_ginsim(m)
ginsim.show(lrg)
```

Out[4]:

Figure 17. Model imported in GINsim, the regulatory graph is depicted.

You can now click inside the space of the fourth command and then click on Insert and choose Insert cell below

```
File Edit View Insert Cell Kernel Widgets Help Non fiable Python 3
```

```
Entrée [3]: M import biolqm
import ginsim

Entrée [4]: M m = biolqm.load("Apoptosis_VS_SSA_AN.sbml")
m = biolqm.sanitize(m)

lrg = biolqm.to_ginsim(m)
ginsim.show(lrg)
```

Load and view the model in GINsim

You will click twice to insert two cells at the end of the notebook

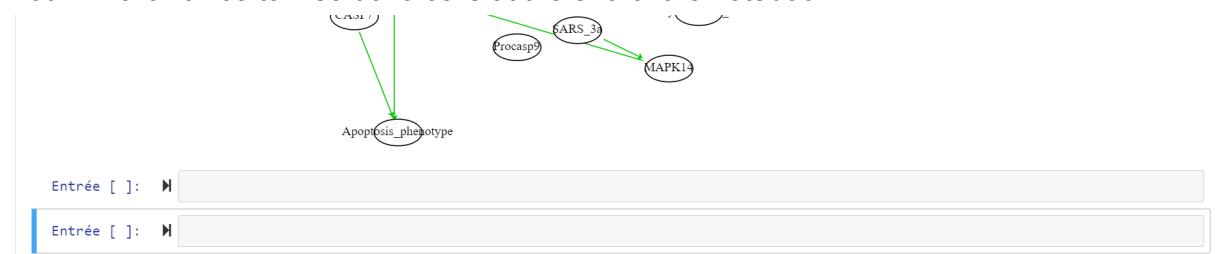


Figure 18. Inserting new cells to the notebook

In the first cell, you will type the following command and execute to obtain the stable states of the system.

Entrée [5]:

```
fps = biolqm.fixpoints(m)
tabulate(fps)
```

Out[5]:

	FASFASL_complex	BIMBADPUMA_complex	Apoptosome_complex	TNFATNFR1_complex	BCL2BCLXLML1_complex	TRADDFADD_complex	TNFR1	F
0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0
...
251	1	1	1	1	1	1	1	1
252	1	1	1	1	1	1	1	1
253	1	1	1	1	1	1	1	1
254	1	1	1	1	1	1	1	1
255	1	1	1	1	1	1	1	1

256 rows × 24 columns

Figure 19. Calculating stable states (fixed points).

and in the second cell you will type:

```
Entrée [6]:
```

```
import pandas as pd
pd.DataFrame(fps).to_csv("fixed_points.csv", index=False)
```

Figure 20. Saving the attractors in a csv file for further processing.

Bonus for the determined!

Exercise 5. Using CaSQ more advanced options from the Python interface

Import both the Apoptosis map and the model you generated earlier on the command-line to the tutorials/CaSQ folder.

Now, using CaSQ's Python API, transform the map into a model but keeping only the components upstream of our phenotype of interest: "Apoptosis_phenotype":

```
In [5]: # disable default logging from casq
casq.logger.disable("casq")

# read the map
info, width, height = casq.read_celldesigner("Curation_Apoptosis_Apoptosis_stable.xml")

# only keep what is upstream of the Apoptosis phenotype
casq.simplify_model(info, ["Apoptosis_phenotype"], [])

# write the corresponding model
casq.write_qual("Curation_Apoptosis_upstream.sbml", info, width, height)
```

Now load and view the original model:

```
In [6]: # load the original model
m1 = biolqm.load("Curation_Apoptosis_Apoptosis_stable.sbml")
m1 = biolqm.sanitize(m1)

lrg1 = biolqm.to_ginsim(m1)
ginsim.show(lrg1)
```

And load and compare with this new restricted model:

```
In [7]: # load and compare the upstream-restricted model
m2 = biolqm.load("Curation_Apoptosis_upstream.sbml")
m2 = biolqm.sanitize(m2)

lrg2 = biolqm.to_ginsim(m2)
ginsim.show(lrg2)
```

Note that the same thing can be done from the command line with the `-u/-upstream` option.

Key notions:

1. XML files of molecular maps encode Process Description diagrams (static)
2. CaSQ compresses the PD diagrams to AF (activity flow) and gives also logical rules that govern the regulation of the components.
3. SIF files contain only the structure of the Boolean models (AF, static) and can be used for topological analysis.
4. SBML qual files contain the executable model (structure plus logical rules) and can be used for *in silico* simulations (dynamic analysis).